INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

Quantified Loop-mu-calculus for Control under Partial Observation

Stéphane Riedweg — Sophie Pinchinat

N° 4949

Mars 2004

THÈME 1

ISSN 0249-6399

Quantified Loop-mu-calculus for Control under Partial Observation

Stéphane Riedweg , Sophie Pinchinat

Thème 1 — Réseaux et systèmes
Projet S4

Rapport de recherche n° 4949 — Mars 2004 — 13 pages

Abstract: In this paper, we consider control problems under partial observation in a logical setting. We extend the mu-calculus by adding to formulas quatifications over atomic propositions and force them inside loop mu-calculus definable classes of [2]. We obtain a very expressive logic having a necessary and sufficient condition on the syntactic normal forms of its formulas to decide their model-checking (and the synthesis of controllers they specify). For example, a maximal permissive controller among a class of controllers under partial observation can be synthesized, as well as decentralized controllers in some cases.

Key-words: Quantified Mu-calculus, quantified loop mu-calculus, control synthsesis, decentrelized controllers, partial observation, optimaly criteria

Unité de recherche INRIA Rennes
IRISA, Campus universitaire de Beaulieu, 35042 Rennes Cedex (France)
Téléphone : +33 2 99 84 71 00 — Télécopie : +33 2 99 84 71 71

Loop-Mu-calcul quantifié pour la synthèse de controlleurs sous observation partielle

Résumé : Dans ce papier, nous abordons le problème du contrôle sous observation partielle d'un point de vue logique. Nous étendons le mu-calcul par quantifications sur les propositions atomiques contraintes à des classes définissables en la logique loop-mu-calcul de [2]. Nous obtenons une logique très expressive qui possède une condition nécessaire et suffisante sur les formes normales des formules pour décider de leurs model-checking (et de synthétiser les contrôleurs qu'elles décrivent). Par exemple, un contrôleur de permissivité maximale parmi les contrôleurs sous observation partielle peut être synthétisé, ainsi que des contrôleurs décentralisés dans certains cas.

Mots-clés : Mu-calcul quantifié, loop- mu-calcul quantifié, synthèse de contrôleurs, contrôle décentralisé, observation partielle, critères d'optimalité

1 Introduction

Recently, [7] introduced the quantified mu-calculus to specify and synthesize controllers for a large class of control problems. This logic consists in quantifications on the atomic propositions of the mu-calculus formulas. A quantified proposition intuitively witnesses the existence of a controller : the proposition is used to mark (on the computation tree of the system) the moves that are allowed under control. Decentralized control can hence be specified as an overlap of two propositions, one for each controller. It is also possible to request a minimal control (or equivalently, the maximal permissivity of the controller). Also, control of open systems as in [4] naturally comes out within the scope of this setting. Unfortunately, since the properties of controllers can be specified only after their synchronization with the plant, there is no way to express unobservability property. This is the main limitation of the whole approach.

In this paper, we extend the quantified mu-calculus to overcome control under partial observation : the quantifications are put into loop mu-calculus classes. The loop mu-calculus was recently introduced by [2] precisely to cope with unobservability of controllers. We then obtain a very expressive logic written $\mathrm{Q}L_\mu^{\circlearrowleft}$ in which e.g. maximal permissivity of a controller among a class of controllers under partial observation can be formulated.

For control synthesis purposes, we study decidability issues for controllers under partial observation, on the basis of a syntactic fragment of $\mathrm{Q}L_\mu^{\circlearrowleft}$ called $\mathrm{Q}L_\mu^{Obs}$. As expected ([2]), the partial observation feature makes the full $\mathrm{Q}L_\mu^{Obs}$ undecidable, preventing from controller synthesis. In this work, we identify the exact syntactic fragment of $\mathrm{Q}L_\mu^{Obs}$ which is decidable. Interestingly, the existence of a maximal permissive controller among a class of controllers under partial observation turns out to be decidable.

The decision procedure (and the synthesis of controllers) relies on the automata counterparts of formulas. Provided some hypothesis on the unobservability sets in the formulas, quotients of automata [2] can hence take partial observation into account. The hypothesis is purely syntactic and identifies the border of decidability. As a matter of fact, decentralized control synthesis becomes decidable whenever the controllers' unobservability sets can be compared.

The paper is organized as follows : Section 2 presents the logical setting. Examples of control problems specifications are given Section 3. The main result on decidability is presented Section 4, and we conclude in Section 5.

2 Quantified loop-mu-calculus

We assume given a finite set of events A, a finite set of propositions AP, and an infinite set of variables $Var = \{X, Y, \dots\}$. In the following, we define the quantified loop mu-calculus ($\mathrm{Q}L_\mu^{\circlearrowleft}$), the loop mu-calculus ($L_\mu^{\circlearrowleft}$) and the mu-calculus (L_μ) where typical elements are α, α_1, resp. $\beta^{\circlearrowleft}, \beta_1^{\circlearrowleft}$, and resp. β, β'.

Definition 1 (Syntax of $\mathrm{Q}L_\mu^{\circlearrowleft}$) *The set of formulas of the quantified loop-mu-calculus on $\Gamma \subseteq AP$, written $\mathrm{Q}L_\mu^{\circlearrowleft}(\Gamma)$, is defined by the grammar:*

$$\exists c \in \beta^{\circlearrowleft} \,.\, \alpha \mid \neg\alpha_1 \mid \alpha_1 \vee \alpha_2 \mid \beta$$

where $c = \{c_a | a \in A\} \subseteq AP$, $\beta^{\circlearrowleft}$ is a formula of the loop-mu-calculus on a set $\Lambda \subseteq AP$ containing c and disjoint from Γ, $\alpha \in \mathrm{Q}L_\mu^{\circlearrowleft}(\Lambda \cup \Gamma)$, α_1 and α_2 are formulas in $\mathrm{Q}L_\mu^{\circlearrowleft}(\Gamma)$, and β is a formula of the pure mu-calculus on Γ.

The loop-mu-calculus defined in [2] rises the pure mu-calculus by adding the atomic loop formulas $\circlearrowleft^a$ for $a \in A$: the set of formulas of the loop- mu-calculus on $\Gamma \subseteq AP$, written $L_\mu^{\circlearrowleft}(\Gamma)$, is defined by the grammar:

$$\top \mid p \mid X \mid \ \circlearrowleft^a \ \mid \neg\beta_1^{\circlearrowleft} \mid \ <a> \beta_1^{\circlearrowleft} \mid \beta_1^{\circlearrowleft} \vee \beta_2^{\circlearrowleft} \mid \mu X.\beta_1^{\circlearrowleft}(X)$$

where $p \in \Gamma$, $X \in Var$, $a \in A$, and the formulas $\beta_1^{\circlearrowleft}$ and $\beta_2^{\circlearrowleft}$ are in $L_\mu^{\circlearrowleft}(\Gamma)$. Moreover, in order to ensure that fix-points formulas have meanings, it is required that in each formula $\mu X.\beta(X)$, X occurs under an even number of negation symbols $\neg$ in $\beta(X)$.
The set of formulas of the pure mu-calculus on Γ, noted $L_\mu(\Gamma)$, is the formulas of the loop-mu-calculus on Γ without any atomic loop formals $\circlearrowleft^a$.
We write $\bot$, $\not\circlearrowleft^a$ $[a]\alpha$, $\alpha_1 \wedge \alpha_2$, $\nu X.\beta(X)$, and $\forall c \in \beta^{\circlearrowleft}.\alpha$ respectively for negating $\top$, $\circlearrowleft^a$, $<a>\neg\alpha$, $\neg\alpha_1 \vee \neg\alpha_2$, $\mu X.\neg\beta(\neg X)$ and $\exists c \in \beta^{\circlearrowleft} \,.\, \neg\alpha$. We write also $\xrightarrow{a}$, $\not\xrightarrow{a}$, and $\alpha_1 \Rightarrow \alpha_2$ respectively for $<a>\top$, $[a]\bot$, and $\neg\alpha_1 \vee \alpha_2$.

We call *sentences* all quantified mu-calculus formulas without free variables.

Remark 1 *The syntax of $\mathrm{Q}L_\mu$ does allow quantifiers only outside fixed-point terms and constraints the quantifiers to range over some loop-mu-calculus classes.*
The quantified-free fragment of $\mathrm{Q}L_\mu^{\circlearrowleft}$ is the pure mu-calculus. The fragment of $\mathrm{Q}L_\mu^{\circlearrowleft}$ without constraint on quantifiers is the quantified mu-calculus defined in [7].

The quantified mu-calculus, as a generalization of the mu-calculus, is also given an interpretation over deterministic transition structures called *processes* in [2] .

Definition 2 *A* process *on $\Gamma \subseteq AP$ is a tuple $\mathcal{S} =< \Gamma, S, s^0, t, L >$, where S is the set of states, $s^0 \in S$ is the initial state, $t : S \times A \to S$ is a partial function called the* transition function *and $L : S \to \mathcal{P}(\Gamma)$ maps states to subset of propositions.*
The process $\mathcal{S}$ is finite *if S is finite and it is* complete *if for all $(a, s) \in A \times S$, $t(s, a)$ is defined.*

Compound processes can be built up by synchronous product.

Definition 3 *The* (synchronous) product *of two processes* $\mathcal{S}_1 =< \Gamma_1, S_1, s_1^0, t_1, L_1 >$ *and* $\mathcal{S}_2 =< \Gamma_2, S_2, s_2^0, t_2, L_2 >$ *on disjoint sets* Γ_1 *and* Γ_2 *is the process* $\mathcal{S}_1 \times \mathcal{S}_2 =< \Gamma, S_1 \times S_2, (s_1^0, s_2^0), t, L >$ *on* $\Gamma = \Gamma_1 \cup \Gamma_2$ *such that:*

- $(s_1', s_2') \in t((s_1, s_2), a)$ *whenever* $s_1' \in t_1(s_1, a)$ *and* $s_2' \in t_2(s_2, a))$,
- $L(s_1, s_2) = L_1(s_1) \cup L_2(s_2)$.

In the sequel, we shall in particular make the product of a process on Γ with another (complete) process on a disjoint set of propositions Λ in order to obtain a similar process on $\Gamma \cup \Lambda$. This is the way in which $\mathrm{Q}L_\mu$ will be applied to solve control problem (see Theorem 2 Section 3).

Definition 4 *A* labeling process on $\Lambda \subseteq AP$ *is simply a complete process* $\mathcal{E}$ *on* Λ. *Now, for any process* $\mathcal{S} =< \Gamma, S, s^0, t, L >$ *with* Γ *disjoint from* Λ, $\mathcal{S} \times \mathcal{E}$ *is called a* labeling of $\mathcal{S}$ (by $\mathcal{E}$) on Λ. *We let* Lab_Λ *denote the set of labeling processes on* Λ.

Notice that labeling $\mathcal{S}$ on $\emptyset$ amounts to unfold process $\mathcal{S}$.
We can now define the semantics of the logic.

Definition 5 (Semantics of $\mathrm{Q}L_\mu^{\circlearrowleft}$**)** *The interpretation of the formulas in* $\mathrm{Q}L_\mu^{\circlearrowleft}(\Gamma)$ *is relative to a process* $\mathcal{S} =< \Gamma, S, s^0, t, L >$ *and a valuation* $val : Var \to \mathcal{P}(S)$. *This interpretation* $[\![\alpha]\!]_{\mathcal{S}}^{[val]} (\subseteq S)$ *is defined inductively as follows.*

- *The interpretation* $[\![\exists c \in \beta^{\circlearrowleft}.\alpha]\!]_{\mathcal{S}}^{[val]}$ *of the quantified formula* $\exists c \in \beta^{\circlearrowleft}.\alpha$, *where* $\beta^{\circlearrowleft} \in L_\mu^{\circlearrowleft}(\Lambda)$ *is the set of states* $s \in S$ *such that there exists a complete process* $\mathcal{E} =< \Lambda, E, \varepsilon^0, t', L' > \in Lab_\Lambda$ *and a valuation* $val' : Var \to \mathcal{P}(E)$, *such that:*

$$(s, \varepsilon^0) \in [\![\alpha]\!]_{\mathcal{S} \times \mathcal{E}}^{[val \times E]} \text{ and } \varepsilon^0 \in [\![\beta^{\circlearrowleft}]\!]_{\mathcal{E}}^{[val']}$$

  where $(val \times E)(X) = val(X) \times E$ *for any* $X \in Var$.

- *the interpretation of the other formulas is defined by:*

$$\begin{array}{l}
[\![\top]\!]_{\mathcal{S}}^{[val]} = S \\
[\![p]\!]_{\mathcal{S}}^{[val]} = \{s \in S \,|\, p \in L(s)\} \\
[\![\circlearrowleft^a]\!]_{\mathcal{S}}^{[val]} = \{s \in S \,|\, t(s, a) = s\} \\
[\![X]\!]_{\mathcal{S}}^{[val]} = val(X) \\
[\![\neg\alpha]\!]_{\mathcal{S}}^{[val]} = S \setminus [\![\alpha]\!]_{\mathcal{S}}^{[val]} \\
[\![<a> \alpha]\!]_{\mathcal{S}}^{[val]} = \{s \in S \,|\, \exists s' : t(s, a) = s' \, and \, s' \in [\![\alpha]\!]_{\mathcal{S}}^{[val]}\} \\
[\![\alpha_1 \vee \alpha_2]\!]_{\mathcal{S}}^{[val]} = [\![\alpha_1]\!]_{\mathcal{S}}^{[val]} \cup [\![\alpha_2]\!]_{\mathcal{S}}^{[val]} \\
[\![\mu X.\alpha(X)]\!]_{\mathcal{S}}^{[val]} = \cap\{V \subseteq S \,|\, [\![\alpha]\!]_{\mathcal{S}}^{[val(V/X)]} \subseteq V\}
\end{array}$$

The valuation val does not influence the semantics of a sentence $\alpha \in \mathrm{Q}L_\mu$*; and we write* $\mathcal{S} \models \alpha$ *whenever the initial state of* $\mathcal{S}$ *belongs to* $[\![\alpha]\!]_{\mathcal{S}}$.

Notice that the set of models of any $\mathrm{Q}L_\mu^{\circlearrowleft}$ formula is closed under bisimulation. We define now $\mathrm{Q}L_\mu^{Obs}$, the fragment of $\mathrm{Q}L_\mu^{\circlearrowleft}$ for control under partial observation requirements, as show in Section 3.

Definition 6 (Syntax of $\mathrm{Q}L_\mu^{Obs}$**)** *The set of formulas on* $\Gamma \subseteq AP$ *of* $\mathrm{Q}L_\mu^{Obs}$ *(written* $\mathrm{Q}L_\mu^{Obs}(\Gamma)$*) is defined by the grammar:*

$$\exists c \in \beta \wedge \mathtt{Obs}(O)\,.\,\alpha \mid \neg\alpha_1 \mid \alpha_1 \vee \alpha_2 \mid \beta'$$

where $c = \{c_a \mid a \in A\} \subseteq AP$*,* O *is a subset of* A*,* $\mathtt{Obs}(O)$ *is the formula* $\nu Y. \bigwedge_{a\in A}[a]Y \wedge \bigwedge_{u\notin O}(\rightarrow^u \Rightarrow \circlearrowleft^u)$*,* β *is a formula of the pure mu-calculus on a set* $\Lambda \subseteq AP$ *containing* c *and disjoint from* Γ*,* $\alpha \in \mathrm{Q}L_\mu^{Obs}(\Lambda \cup \Gamma)$*,* α_1 *and* α_2 *are formulas in* $\mathrm{Q}L_\mu^{Obs}(\Gamma)$*, and* β' *is a formula of the pure mu-calculus on* Γ.

3 Control under partial observation

This section presents various examples of control requirements in $\mathrm{Q}L_\mu^{Obs}$. First, a transformation of formulas is defined, which used to link $\mathrm{Q}L_\mu^{Obs}$ with controllers under partial observation, as shown by Theorem 2. Variants of the Theorem are then exploited to capture requirements, such as *maximally permissive controller among the controllers under partial observation* and *decentralized controllers.*

Definition 7 *For any sentence* $\alpha \in \mathrm{Q}L_\mu^{Obs}(\Gamma)$*, and for any* $x = \{x_a; a \in A\} \subseteq AP$*, the* x-lift of α *is the formula* $\alpha * x \in \mathrm{Q}L_\mu^{Obs}(\Gamma \cup x)$*, inductively defined by :*

$\top * x = \top$	$p * x = p$
$X * x = X$	$(\neg\alpha) * x = \neg(\alpha * x)$
$(\alpha_1 \vee \alpha_2) * x = \alpha_1 * x \vee \alpha_2 * x$	$(<a>\,\beta) * x = x_a \wedge <a>\,(\beta * x)$
$(\mu X.\beta) * x = \mu X.(\beta * x)$	$(\exists c \in \beta \wedge \mathtt{Obs}(O).\alpha) * x = \exists c \in \beta \wedge \mathtt{Obs}(O).(\alpha * x)$

Definition 8 *Given a process* $\mathcal{S} =< \Gamma, S, s^0, t, L >$ *and given* $\{x_a; a \in A\} \subseteq \Gamma$*; the* x-pruning of $\mathcal{S}$ *is the process* $\mathcal{S}_{(\rightarrow^x)} =< \Gamma, S, s^0, t', L >$ *such that, for all* $s \in S$ *and* $a \in A$*,* $t'(s,a) = t(s,a)$ *if* $x_a \in L(s)$ *or* $t'(s,a)$ *is undefined otherwise.*

Lemma 1 *For any process* $\mathcal{S}$ *on* Γ*, for any* $x = \{x_a \mid a \in A\} \subseteq \Gamma$*, and for any sentence* $\alpha \in \mathrm{Q}L_\mu^{Obs}(\Gamma)$*, we have:* $[\![\alpha]\!]_{\mathcal{S}_{(\rightarrow^x)}} = [\![\alpha * x]\!]_{\mathcal{S}}$.

Proof Straightforward by induction on α. □

Control synthesis [6, 2, 3, 8] is a generic problem that consists, given a process $\mathcal{S}$ called a plant and a property α, to enforce this property by making the product of the plant with an additional control system $\mathcal{R}$ that restricts its behaviors. $\mathcal{R}$ is called a *controller on* $\mathcal{S}$ *for* α, and the goal is to synthesize such controller. We focus on the class of control under partial observation [2, 3]: for a given partition of A into unobservable events U and observable events O; the controller $\mathcal{R}$ must satisfy the loop-mu-calculus formula $\mathtt{Obs}(O)$.

Theorem 2 *Given a process* $\mathcal{S}$ *on* Γ*, a set* $O \subseteq A$*, and two sentences* $\alpha \in \mathrm{Q}L_\mu^{Obs}(\Lambda \cup \Gamma)$ *and* $\beta \in L_\mu(\Lambda)$ *where* Λ *and* Γ *are disjoint, the following assertions are equivalent :*

- *There is a controller* $\mathcal{R}$ *satisfying* $\beta \wedge \mathtt{Obs}(O)$ *such that* $\mathcal{S} \times \mathcal{R} \models \alpha$
- $\mathcal{S} \models \exists c \in \beta * c \wedge \mathtt{Obs}(O).\alpha * c$ *where* $c = \{c_a \mid a \in A\}$ *is a set of fresh propositions.*

Proof First, suppose that there exists a process $\mathcal{R}$ on Λ satisfying $\mathtt{Obs}(O) \wedge \beta$ and such that $\mathcal{S} \times \mathcal{R} \models \alpha$. Given $c = \{c_a; a \in A\} \subseteq AP \setminus (\Lambda \cup \Gamma)$, we can easily define a labeling process $\mathcal{E} \in Lab_{\Lambda \cup c}$ satisfying $\mathtt{Obs}(O)$ and such that $(\mathcal{E})_{(\rightarrow^c)}$ is the process $\mathcal{R}$ without c. Now, $(\mathcal{S} \times \mathcal{E})_{(\rightarrow^c)}$ or equivalently $\mathcal{S} \times (\mathcal{E})_{(\rightarrow^c)}$ satisfies α, since $\mathcal{R}$ is $(\mathcal{E})_{(\rightarrow^c)}$ without c and c does not occur in α. Using Lemma 1, we conclude that $(\mathcal{S} \times \mathcal{E}) \models \alpha * c$ and that $\mathcal{E} \models \beta * c$. Suppose now that $\mathcal{S} \models \exists c \in \beta * c \wedge \mathtt{Obs}(O).\alpha * c$. By Definition 5, there is a labeling process $\mathcal{E} \in Lab_{\Lambda \cup c}$ such that $\mathcal{E} \models \mathtt{Obs}(O) \wedge \beta * c$ and $\mathcal{S} \times \mathcal{E} \models \alpha * c$. Then $\mathcal{E}_{(\rightarrow^c)} \models \mathtt{Obs}(O)$ and by Lemma 1, $\mathcal{E}_{(\rightarrow^c)}$ satisfies β and $(\mathcal{S} \times \mathcal{E})_{(\rightarrow^c)}$ satisfies α. Take $\mathcal{R}$ as $(\mathcal{E})_{(\rightarrow^c)}$ to conclude. □

We illustrate the use of $\mathrm{Q}L_\mu^{Obs}$ to specify control requirements. The x-lift is extended to any conjunctions by $\alpha * (x_1 \wedge x_2) = (\alpha * x_1) * x_2$. We let $\mathtt{Inv}(\beta) = \nu Y. \bigwedge_{a \in A} [a] Y \wedge \beta$ and $\mathtt{Reach}(\beta) = \mu X. \bigvee_{a \in A} \langle a \rangle X \vee \beta$.

Decentralized controllers for α. The formula (1) expresses the existence of two controllers $\mathcal{R}_1$ and $\mathcal{R}_2$ on $\mathcal{S}$ such that $\mathcal{S} \times \mathcal{R}_1 \times \mathcal{R}_2 \models \alpha$; and for each $i = 1, 2$, $\mathcal{R}_i$ can observes only the set of events $O_i \subseteq A$, and can disables only transitions labeled by a (controllable) event in $C_i \subseteq A$.

$$\begin{array}{rl} \mathcal{S} \models & \exists c_1 \in \mathtt{Obs}(O_1) \wedge (\mathtt{Inv}(\bigwedge_{uc \notin C_1} \rightarrow^{uc})) * c_1. \\ & \exists c_2 \in \mathtt{Obs}(O_2) \wedge (\mathtt{Inv}(\bigwedge_{uc \notin C_2} \rightarrow^{uc})) * c_2.\alpha * (c_1 \wedge c_2) \end{array} \quad (1)$$

Maximally permissive controller among the controllers under partial observation for α. A controller c for α is *maximally permissive* if no other controller c' for α can cut strictly less transitions than c. Writing $c \subsetneq c'$ the mu-calculus formula $(\mathtt{Inv}(\bigwedge_{a \in A} c_a \Rightarrow c'_a)) * c \wedge (\mathtt{Reach}(\bigvee_{a \in A} \neg c_a \wedge c'_a)) * c'$; this requirement is expressed by the formula (2)

$$\mathcal{S} \models \exists c \in \mathtt{Obs}(O).\alpha * c \wedge [\forall c' \in \mathtt{Obs}(O).c \subsetneq c' \Rightarrow \neg(\alpha * c')] \quad (2)$$

4 Deciding $\mathrm{Q}L_\mu^{Obs}$

This section is dedicated to the model-checking of $\mathrm{Q}L_\mu^{Obs}$. With Theorem 3; we present a sufficient and necessary condition to decide the model-checking and when the condition holds, Theorem 6 is used to synthesize controllers. This condition is based on the order of formulas $\mathtt{Obs}(O)$ occurring in normal form of sentences (Definition 9).

Definition 9 (Normal form) *The set of sentences of* $\mathrm{Q}L_\mu^{Obs}(\Gamma)$ *in* normal form *is the set of sentences:*

$$Q_1 c_1 \in \beta_1 \wedge \mathtt{Obs}(O_1) \wedge \ldots Q_n c_n \in \beta_n \wedge \mathtt{Obs}(O_n).\beta$$

where for $i \in \{1,\ldots,n\}$*:* $Q_i \in \{\exists,\forall\}$*,* $\beta_i \in L_\mu(\Lambda_i)$*, and* $\beta \in L_\mu(\Gamma \cup \Lambda_1 \cup \cdots \cup \Lambda_n)$*. Moreover, for all* $i \in \{1,\ldots,n-1\}$*;* $O_i \neq \emptyset$*,* $O_{i+1} \neq \emptyset$ *and* $Q_i = Q_{i+1}$ *imply that* O_{i+1} *is not a proper subset of* O_i*.*

It can obviously be shown that each sentence of $\mathrm{Q}L_\mu^{Obs}$ can be written in normal form (quantifiers of the same type commute).

Theorem 3 (Main result). *Let* $\alpha \in \mathrm{Q}L_\mu^{Obs}(\Gamma)$ *be the sentence (in normal form)* $Q_1 c_1 \in \beta_1 \wedge \mathtt{Obs}(O_1).\ldots.Q_n c_n \in \beta_n \wedge \mathtt{Obs}(O_n).\beta$*.*
The problem to check whether a process $\mathcal{S}$ *satisfies* α *is decidable if and only if* (C)*: the subsequence of* $(O_i)_{1\leq i\leq n}$ *of nonempty sets is increasing.*

4.1 Decidable fragment

We use the two main constructions of [2] to show that the condition (C) of Theorem 3 is sufficient. Theses constructions are given in [2] as automata constructions. Proposition 4 and Proposition 5 give their logical version since the mu-calculus and the parity automata as well as the loop-mu-calculus and the parity loop automata are equivalent [2, 1].

Proposition 4 (Quotient of $L_\mu^{\circlearrowleft}$ over process) *Let* $\mathcal{S}$ *be a process on* Γ*, let* $\beta^{\circlearrowleft} \in L_\mu^{\circlearrowleft}(\Gamma \cup \Lambda)$ *be a sentence and suppose that* Λ *and* Γ *are disjoint sets.*
There exists a sentence $\beta^{\circlearrowleft}/\mathcal{S}$ *of the loop-mu-calculus on* Γ*, called the quotient of* $\beta^{\circlearrowleft}$ *over* $\mathcal{S}$*, such that, for any process* $\mathcal{P}$ *on* Γ*, we have:*

$$\mathcal{P} \models \beta^{\circlearrowleft}/\mathcal{S} \text{ if and only if } \mathcal{P} \times \mathcal{S} \models \beta^{\circlearrowleft}$$

Proposition 5 (Quotient of $L_\mu^{\circlearrowleft}$ over L_μ) *Let* $\beta \in L_\mu(\Lambda)$ *and let* $\beta^{\circlearrowleft} \in L_\mu^{\circlearrowleft}(\Gamma \cup \Lambda)$ *be sentences and suppose that* Λ *and* Γ *are disjoint sets.*
There exists a sentence $\beta^{\circlearrowleft}/\beta$ *of the loop-mu-calculus on* Γ*, called the quotient of* $\beta^{\circlearrowleft}$ *over* β*, such that, for any process* $\mathcal{P}$ *on* Γ*, we have:*

$$\mathcal{P} \models \beta^{\circlearrowleft}/\beta \text{ if and only if } \exists \mathcal{Q} \text{ s.t. } \mathcal{Q} \models \beta \text{ and } \mathcal{P} \times \mathcal{Q} \models \beta^{\circlearrowleft}$$

Notice that in Proposition 5, the divisor must be a mu-calculus sentence. Hence, this quotient can be used to constraint each quantifier to be put into some mu-calculus class, but not into a loop-mu-calculus class. We can nevertheless enforce some loop constraints on the process $\mathcal{Q}$ of Proposition 5 inside $\beta^{\circlearrowleft}$, as shown by Theorem 6.

Theorem 6 (Decidable fragment (and synthesis)). *The model checking of the sentences in* $\mathrm{Q}L_\mu^{Obs}$ *is decidable whenever the condition* (C) *of Theorem 3 holds.*

Proof Let $\mathcal{S}$ be a process on Γ and let $\alpha \in \mathrm{Q}L_\mu^{Obs}(\Gamma)$ be a sentence in normal form satisfying the condition (C) of Theorem 6. We will construct a loop-mu-calculus sentence $\beta^{\circlearrowleft} \in L_\mu^{\circlearrowleft}(\Gamma)$ such that $\beta^{\circlearrowleft}$ has a model if and only if $\mathcal{S}$ satisfies α. Moreover, a model of $\beta^{\circlearrowleft}$ (if any) is a labeling process corresponding to c_1.

Let $(\alpha_i)_{1\leq i\leq n}$ be the sequence of $\mathrm{Q}L_\mu^{Obs}$-sentences such that:

$$\begin{cases} \alpha = \exists c_1 \in \beta_1 \wedge \mathtt{Obs}(O_1).\alpha_1 \\ \alpha_i = Q_{i+1}c_{i+1} \in \beta_{i+1} \wedge \mathtt{Obs}(O_{i+1}).\alpha_{i+1} \quad \text{for all } i \in \{1,\ldots,n-1\} \\ \alpha_n = \beta \end{cases}$$

We can suppose that $Q_1 = \exists$ and that the sequence $(O_i)_{1\leq i\leq n}$ is increasing since, for any arbitrary $O \subseteq A$, a sentence of the form $Qc \in \mathtt{Obs}(\emptyset) \wedge \beta'.\alpha'$ is equivalent to the sentence $Qc \in \mathtt{Obs}(O) \wedge \beta' \wedge \bigwedge_{p\in AP}(\mathtt{Inv}(p) \vee \mathtt{Inv}(\neg p)).\alpha'$. Let $\mathtt{Full}$ be the mu-calculus sentence defining the class of complete process; we define the sentence $\beta^{\circlearrowleft} \in L_\mu^{\circlearrowleft}(\Gamma)$ inductively by:

$$\begin{cases} \beta^{\circlearrowleft} = \mathtt{Obs}(O_1) \wedge \beta_1 \wedge \mathtt{Full} \wedge \beta_1^{\circlearrowleft} \\ \beta_i^{\circlearrowleft} = \begin{cases} \big(\beta_{i+1}^{\circlearrowleft} \wedge \mathtt{Obs}(O_{i+1})\big)/\big(\beta_{i+1} \wedge \mathtt{Full}\big) & \text{if} \quad Q_{i+1} = \exists (1 \leq i \leq n-1) \\ \neg\big((\neg \ \beta_{i+1}^{\circlearrowleft} \wedge \mathtt{Obs}(O_{i+1}))/(\beta_{i+1} \wedge \mathtt{Full})\big) & \text{if} \quad Q_{i+1} = \forall (1 \leq i \leq n-1) \end{cases} \\ \beta_n^{\circlearrowleft} = \beta/\mathcal{S} \end{cases}$$

Lemma 7 *Let* $\{\mathcal{E}_i | \, i = 1,\ldots,n\}$ *be a set of labeling process.*
For any $k \in \{1,\ldots,n\}$, *if* $\Pi_{i=1}^k \mathcal{E}_i$ *satisfies the formula* $\mathtt{Obs}(O_k)$, *we have:*

$$\mathcal{S} \times \Pi_{i=1}^k \mathcal{E}_i \models \alpha_k \Leftrightarrow \Pi_{i=1}^k \mathcal{E}_i \models \beta_k^{\circlearrowleft}$$

Proof (of Lemma 7) by induction on $k' = n - k$.
The basic case where $k' = 0$ (hence $k = n$) is obtained by Proposition 4. Suppose now that the assertion of Lemma 7 holds for the rank k', we demonstrate the assertion for the rank $k' + 1$:

Suppose that $\Pi_{i=1}^{k-1} \mathcal{E}_i \models \mathtt{Obs}(O_{k-1})$. We must show the equivalence between $\mathcal{S} \times \Pi_{i=1}^{k-1} \mathcal{E}_i \models \alpha_{k-1}$ and $\Pi_{i=1}^{k-1} \mathcal{E}_i \models \beta_{k-1}^{\circlearrowleft}$. Consider the case where $Q_k = \exists$, the case $Q_k = \forall$ is

similar. We have:

$$S \times \Pi_{i=1}^{k-1}\mathcal{E}_i \models \alpha_{k-1} \overset{(def)}{\Longleftrightarrow} \exists \mathcal{E}_k \in Lab_{\Lambda_k} \text{ s.t. } \begin{cases} \mathcal{E}_k \models \beta_k \wedge \mathtt{Obs}(O_k) \\ \text{and} \\ S \times \Pi_{i=1}^{k}\mathcal{E}_i \models \alpha_k \end{cases}$$

$$\overset{(i.h.)}{\Longleftrightarrow} \exists \mathcal{E}_k \in Lab_{\Lambda_k} \text{ s.t. } \begin{cases} \mathcal{E}_k \models \beta_k \wedge \mathtt{Obs}(O_k) \\ \text{and} \\ \Pi_{i=1}^{k}\mathcal{E}_i \models \beta_k^{\circlearrowleft} \end{cases}$$

By hypothesis, we have: $\Pi_{i=1}^{k-1}\mathcal{E}_i \models \mathtt{Obs}(O_{k-1})$ and $O_{k-1} \subseteq O_k$. Then $\mathcal{E}_k$ satisfies $\mathtt{Obs}(O_k)$ if and only if the product $\Pi_{i=1}^{k}$ satisfies $\mathcal{E}_i \models \mathtt{Obs}(O_k)$. Also, we have the following equivalence:

$$S \times \Pi_{i=1}^{k-1}\mathcal{E}_i \models \alpha_{k-1} \Leftrightarrow \exists \mathcal{E}_k \in Lab_{\Lambda_k} \text{ s.t. } \begin{cases} \mathcal{E}_k \models \beta_k \\ \text{and} \\ \Pi_{i=1}^{k-1}\mathcal{E}_i \times \mathcal{E}_k \models \beta_k^{\circlearrowleft} \wedge \mathtt{Obs}(O_k) \end{cases}$$

Using Proposition 5, we have $S \times \Pi_{i=1}^{k-1}\mathcal{E}_i \models \alpha_{k-1}$ if and only if $\Pi_{i=1}^{k-1}\mathcal{E}_i$ satisfies $(\beta_k^{\circlearrowleft} \wedge \mathtt{Obs}(O_k))/(\beta_k \wedge \mathtt{Full})$; which is the formula $\beta_{k-1}^{\circlearrowleft}$. □

We turn back to the proof of Theorem 6. We have the two assertions:

- $\mathcal{S} \models \alpha$ if and only if there exists a process $\mathcal{E}_1 \in Lab_{\Lambda_1}$ such that: $\mathcal{E}_1 \models \mathtt{Obs}(O_1) \wedge \beta_1$ and $\mathcal{S} \times \mathcal{E}_1 \models \alpha_1$
- $\beta^{\circlearrowleft}$ has a model if and only if there exists a process $\mathcal{E}_1$ such that: $\mathcal{E}_1 \models \mathtt{Obs}(O_1) \wedge \beta_1 \wedge \mathtt{Full}$ and $\mathcal{E}_1 \models \beta_1^{\circlearrowleft}$.

Using Lemma 7, we conclude that $\mathcal{S} \models \alpha$ if and only if $\beta^{\circlearrowleft}$ has a model. Using loop-automata of [2], we can both decide and synthesize.

□

4.2 Undecidable fragment

We show here that the condition (C) of Theorem 3 is necessary. We consider the formulas $Q_1c_1 \in \mathtt{Obs}(O_1) \wedge \beta_1.Q_2c_2 \in \mathtt{Obs}(O_2) \wedge \beta_2.\beta$ of $\mathrm{Q}L_\mu^{Obs}$ (where $\beta \in L_\mu$) such that Condition (C) does not hold. The case where O_1 and O_2 are uncomparable is done by Theorem 8 and the case where O_2 is a nonempty proper subset of O_1 and Q_1 and Q_2 are different is done by Theorem 9.

Theorem 8 *The model-checking of the formulas in normal form*

$$Q_1c_1 \in \mathtt{Obs}(O_1) \wedge \beta_1.Q_2c_2 \in \mathtt{Obs}(O_2) \wedge \beta_2.\beta$$

is undecidable whenever $O_1 \setminus O_2$ and $O_2 \setminus O_1$ are nonempty sets.

Proof By an adaptation of [2], the Post correspondence problem is reduced to model-check such a formula.
We consider a Post system, that pairs of words $\{(u_i, v_i)|i = 1, \ldots, n\}$ over some alphabet Σ. The Post correspondance problem is to find a sequence $i_1, i_2, ..., i_k$ of selections such that the words $u_{i_1}u_{i_2}...u_{i_k}$ and $v_{i_1}v_{i_2}...v_{i_k}$ formed by concatenation are identical.

We first show first Theorem 8 for the case where $Q_1 = Q_2 = \exists$. As explained in [2], the correspondence problem has a solution if and only if there exist two finite nonempty words $x \in \Sigma^*$ and $y \in D^*$ (for $D = \{\$_i|i = 1, \ldots, n\}$ the set that marks the n different dominoes of the Post system) such that the shuffle of x and y has a nonempty intersection with both sets $L_1 = \{\$_i u_i|i = 1, \ldots, n\}^*$ and $L_2 = \{\$_i v_i|i = 1, \ldots, n\}^*$. The idea is next to code the words x and y resp. by processes $\mathcal{P}_x$ on Σ and $\mathcal{P}_y$ on D such that the set of paths in $\mathcal{P}_x \times \mathcal{P}_y$ is the shuffle of x and y.

We adapt the proof of [2] to our framework by using the sets Σ and D as propositions instead of letters. Our coding is the following. We use the set of actions $\{\sigma, \$\} \subseteq A$: $e \in \Sigma$ is modeled by a σ-transition in $\mathcal{P}_x$ toward a state labeled by e. Similarly, $\$_i \in D$ is modeled by a \$-transition in $\mathcal{P}_y$ toward a state labeled by $\$_i$. In this way, following [2], it can be shown that the Post correspondence problem has a solution if and only if there exist two processes $\mathcal{P}_x$ and $\mathcal{P}_y$, such that: $\mathcal{P}_x \models \mathtt{Obs}(\{\sigma\}) \wedge \beta_1$, $\mathcal{P}_y \models \mathtt{Obs}(\{\$\}) \wedge \beta_2$ and $\mathcal{P}_x \times \mathcal{P}_y \models \mathtt{Post}$, where β_1, β_2 and $\mathtt{Post}$ are formally defined below (take $\# = \not\to^\sigma \wedge \not\to^\$$, and for any word w, $|w|$ is its length and $w(i)$ is its i'th letter).

$$\begin{aligned}
\beta_1 &\equiv \mathtt{Inv}(\to^\$) \wedge \langle\sigma\rangle\, (\mathtt{Inv}(\textstyle\bigvee_{e_i \in \Sigma}(e_i \wedge \bigwedge_{e_j \neq e_i} \neg e_j))) \wedge \mu X.\, \langle\sigma\rangle\, X \vee \not\to^\sigma \\
\beta_2 &\equiv \mathtt{Inv}(\to^\sigma) \wedge \langle\$\rangle\, (\mathtt{Inv}(\textstyle\bigvee_{\$_i \in \$}(\$_i \wedge \bigwedge_{j \neq i} \neg \$_j))) \wedge \mu X.\, \langle\$\rangle\, X \vee \not\to^\$ \\
\mathrm{Post} \equiv\ & \mu X. \textstyle\bigvee_{i \in \{1,..,n\}} \langle\$\rangle \Big(\$_i \wedge \langle\sigma\rangle \big(u_i(1) \wedge \langle\sigma\rangle\, (\ldots \langle\sigma\rangle\, (u_i(|u_i|) \wedge (X \vee \#)..)\Big) \\
\wedge\ & \mu X. \textstyle\bigvee_{i \in \{1,..,n\}} \langle\$\rangle \Big(\$_i \wedge \langle\sigma\rangle \big(v_i(1) \wedge \langle\sigma\rangle\, (\ldots \langle\sigma\rangle\, (v_i(|v_i|) \wedge (X \vee \#)..)\Big)
\end{aligned}$$

Now, the Post correspondence problem has a solution if and only if, for a given complete process $\mathcal{S}$, we have:
$\mathcal{S} \models \exists c_1 \in \mathtt{Obs}(\{\sigma\}) \wedge \beta_1 * c_1 . \exists c_2 \in \mathtt{Obs}(\{\$\}) \wedge \beta_2 * c_2 . \mathtt{Post} * (c_1 \wedge c_2)$

The proof for $Q_1 = \forall$ and $Q_2 = \exists$ consists in checking absence of solution for the Post correspondance problem : for all word $y = \$_{i_1}\$_{i_2}...\$_{i_k} \in D^*$, the corresponding word $u_{i_1}u_{i_2}...u_{i_k}$ differs from $v_{i_1}v_{i_2}...v_{i_k}$. The other cases for Q_1 and Q_2 are obtained by negation. □

Theorem 9 *The model-checking of the formulas*

$$\exists c_1 \in \mathtt{Obs}(O_1) \wedge \beta_1 . \forall c_2 \in \mathtt{Obs}(O_2) \wedge \beta_2 . \beta$$

is undecidable if O_2 is a nonempty proper subset of O_1.

Proof Let $o_1 \in O_1 \setminus O_2$ and let $o_2 \in O_2$. We reduce the Tiling problem (see [5]) to model-check such a formula. Consider the infinite grid $\mathbb{N}^2$ of couples of positive integers and a finite set of tiles $T = \{t_1, \ldots t_n\}$. Let V and H be two relations on T: $V \subseteq T \times T$

and $H \subseteq T \times T$, called respectively vertical compatibility and horizontal compatibility. The Tiling problem is to label each node $(i,j) \in I\!N^2$ of the grid by a unique tile $t(i,j) \in T$ while respecting the compatibilities : namelly, $(t(i,j), t(i,j+1)) \in H$ and $(t(i,j), t(i+1,j)) \in V$. This problem is known to be undecidable (see [5]).

Let $\mathcal{S}$ be the process with only three transitions $(1, o_1, 1)$, $(1, o_2, 2)$, $(2, o_2, 2)$. A node $(n,m) \in I\!N^2$ of the grid is reached by a sequence of actions which corresponds to the state ${o_1}^n.{o_2}^m$ in the unfolding of $\mathcal{S}$. (Notice that not all the paths in the grid are represented in this unfolding).

Hence, the Tiling problem has a solution if and only if we can label each state ${o_1}^n.{o_2}^m$ by a unique tile $t({o_1}^n.{o_2}^m) \in T$ such that Conditions (3) and (4) (for compatibilty) hold:

$$(t({o_1}^n.{o_2}^m), t({o_1}^n.{o_2}^{m+1})) \in V \tag{3}$$

$$(t({o_1}^n.{o_2}^m), t({o_1}^{n+1}.{o_2}^m)) \in H \tag{4}$$

In other words, the Tiling problem is reduced to find a labeling process $\mathcal{E}_1$ on $T \cup c_1$, which defines a "tile-mapping" t satisfying Conditions (3) and (4).

In the rest of the section we show how to express the existence of $\mathcal{E}_1$ in our logic. First, $\mathcal{E}_1$ chooses a tile-mapping. This is expresses by $\mathtt{Obs}(\{o_1, o_2\}) \wedge \beta_1 * c_1$, where $\beta_1 \in L_\mu$ states that exactely one tile is put on each $o_1^n.o_2^m$:

$$\beta_1 \equiv \mathtt{Inv}(\rightarrow^{o_1} \wedge \rightarrow^{o_2}) \wedge \mathtt{Inv}(\bigvee_{i \in \{1,\ldots,n\}} (t_i \wedge \bigwedge_{j \neq i} \neg t_j))$$

Next, we express that Condition (3) and the case $m = 0$ of Condition (4) hold for this choice: $\mathcal{S} \times \mathcal{E}_1$ must satisfy the formula $\mathtt{Tiling(1)} * c_1$ stating that the tile-mapping is indeed compatible along paths in $\mathcal{S}$. Since, Condition (4) for the case where $m > 0$ cannot be expressed this way, it is considered apart below.

$$\mathtt{Tiling(1)} \equiv \mathtt{Inv}(\bigvee_{(t,t') \in V} t \Rightarrow <o_2> t') \wedge \mathtt{Inv}(\bigvee_{(t,t') \in H} t \Rightarrow <o_1> t')$$

Finally, Condition (4) for the remaining cases where $m > 0$ is expressed as follows: $\mathcal{E}_2 \in Lab_{c_2}$ is a labeling process which chooses some $m > 0$ and marks the set of states $\{o_1^n.o_2^m | n \in I\!N\}$. Horizontal compatibility along the horizontal line $(0,m) \rightarrow (1,m) \rightarrow ...(n,m) \rightarrow ..$ of the grid is rephrased in $\mathcal{S} \times \mathcal{E}_1 \times \mathcal{E}_2$ by $\mathtt{Tiling(2)}$:

$$\mathtt{Tiling(2)} \equiv \nu Y.[o_1]Y \wedge \bigvee_{(t,t') \in H} \mu X.\ <o_2>\ (X \vee \not\rightarrow^{o_2} \wedge t) \wedge <o_1>\ (\mu X.(<o_2>\ X \vee \not\rightarrow^{o_2} \wedge t'))$$

Now, the Tiling problem has a solution if and only if

$$\mathcal{S} \models \exists c_1 \in \mathtt{Obs}(\{o_1, o_2\}) \wedge \beta_1 * c_1 . \forall c_2 \in \mathtt{Obs}(\{o_2\}) \wedge \beta_2 * c_2 .\ \mathtt{Tiling(1)} * c_1 \wedge \mathtt{Tiling(2)} * (c_1 \wedge c_2)$$

□

5 Conclusion

The contribution presents a logical setting to specify and synthesize a large class of controllers under partial observation. As it is, when several controllers are combined, the approach based on $\mathrm{Q}L_\mu^{Obs}$ focuses on their synchronous product, given by the conjunction of the labelings. By considering disjunctions of labelings inside the more general $\mathrm{Q}L_\mu^{\bigcirc}$-formulas would enable us to simulate the asynchronous product of controllers under partial observation, but then loosing Condition (C) of Theorem 3 onto which any hope of effectiveness relies.

References

[1] Arnold, A. and D. Niwinski, "Rudiments of mu-calculus," Elsevier, 2001.

[2] Arnold, A., A. Vincent and I. Walukiewicz, *Games for synthesis of controllers with partial observation*, Theoretical Computer Science **303** (2003), pp. 7–34.
URL `http://www.labri.fr/Perso/ igw/Papers/igw-synthesis.ps`

[3] Bergeron, A., *A unified approach to control problems in discrete event processes*, Theoretical Informatics and Applications **27** (1993), pp. 555–573.

[4] Kupferman, O., M. Y. Vardi and P. Wolper, *An automata-theoretic approach to branching-time model checking*, Journal of the ACM **47** (2000), pp. 312–360.

[5] Lewis, H. and C. Papadimitriou, "Elements of the Theory of Computation," Prentice Hall, 1981.

[6] Ramadge, P. J. and W. M. Wonham, *The control of discrete event systems*, Proceedings of the IEEE; Special issue on Dynamics of Discrete Event Systems **77** (1989), pp. 81–98.

[7] Riedweg, S. and S. Pinchinat, *Quantified mu-calculus for control synthesis*, in: *Mathematical Foundations of Computer Science 2003, LNCS 2747* (2003), pp. 642–651.

[8] Vincent, A., *Synthèse de contrôleurs et stratégies gagnantes dans les jeux de parité*, in: Hermès, editor, *Modélisation des systèmes réactifs*, 2001, pp. 87–98.

Unité de recherche INRIA Rennes
IRISA, Campus universitaire de Beaulieu - 35042 Rennes Cedex (France)

Unité de recherche INRIA Futurs : Parc Club Orsay Université - ZAC des Vignes
4, rue Jacques Monod - 91893 ORSAY Cedex (France)
Unité de recherche INRIA Lorraine : LORIA, Technopôle de Nancy-Brabois - Campus scientifique
615, rue du Jardin Botanique - BP 101 - 54602 Villers-lès-Nancy Cedex (France)
Unité de recherche INRIA Rhône-Alpes : 655, avenue de l'Europe - 38334 Montbonnot Saint-Ismier (France)
Unité de recherche INRIA Rocquencourt : Domaine de Voluceau - Rocquencourt - BP 105 - 78153 Le Chesnay Cedex (France)
Unité de recherche INRIA Sophia Antipolis : 2004, route des Lucioles - BP 93 - 06902 Sophia Antipolis Cedex (France)

Éditeur
INRIA - Domaine de Voluceau - Rocquencourt, BP 105 - 78153 Le Chesnay Cedex (France)
http://www.inria.fr
ISSN 0249-6399